

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-190491

(43)Date of publication of application : 23.07.1996

(51)Int.Cl

G06F 9/46

(21)Application number : 07-015552

(71)Applicant : HITACHI KOKI HARAMACHI CO LTD

(22)Date of filing : 04.01.1995

(72)Inventor : KUMAGAI KATSUMI
JINME YOSHIAKI

(54) TASK SWITCHING METHOD

(57)Abstract:

PURPOSE: To shorten the time required to register an execution request task and to determine a task to be executed, and to easily control the tasks, by determining the order of the tasks to be executed in advance.

CONSTITUTION: Task data are data showing information on a task and consist of a state flag indicating that the task is being executed, being interrupted, or being suspended, restart priority data showing the priority when the execution of the task is restarted, a register value when the task is started, restart data showing an address value, etc., and a task status indicating whether or not there is a request to execute the task. In an execution order table 25, the names of tasks to be executed are registered corresponding to counter values 1, 2, 3...Cn. In a task counter, one of the counter values 1-Cn is stored and the task corresponding to the counter value stored in the task counter is a top-priority candidate for a task to be executed next.

タスクカウンタ のカウント値	実行タスク名
1	タスク1
2	タスク2
3	タスク3
.	.
.	.
Cn	タスクn

~ 25

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-190491

(43) 公開日 平成8年(1996)7月23日

(51) Int.Cl.⁸

G 0 6 F 9/46

識別記号

3 4 0 E

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数2 F D (全8頁)

(21) 出願番号 特願平7-15552

(22) 出願日 平成7年(1995)1月4日

(71) 出願人 000153513

株式会社日立工機原町

福島県原町市北長野字南原田70番地

(72) 発明者 熊谷 克己

茨城県ひたちなか市武田1060番地 日立工機 株式会社内

(72) 発明者 甚目 良明

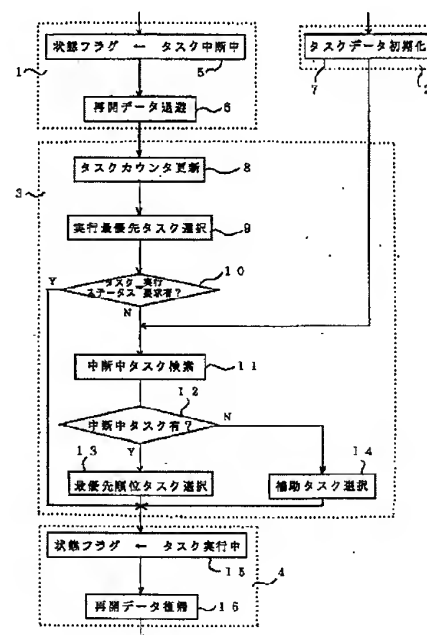
茨城県ひたちなか市武田1060番地 日立工機 株式会社内

(54) 【発明の名称】 タスク切り換え方法

(57) 【要約】

【目的】 本発明の目的は、複数のタスクを時分割により順次実行するリアルタイム処理において、タスクの実行要求、実行するタスクの決定に要する時間を短縮するとともに実行優先順位が同等レベル以下のタスクの実行も可能にすることである。

【構成】 複数のタスクデータ21が予め作成登録されるとともにタスクの実行順序が予め登録された実行順序テーブル25を備え、実行順序テーブル25に登録されているタスクの実行順序に従い実行するタスクを決定する際に、順番の回ってきたタスクのタスクデータ21中のタスクステータス20を検査し、実行可能であれば該タスクの実行を行うようにしたタスク切り換え方法。



【特許請求の範囲】

【請求項 1】 複数のタスクを時分割により順次実行するリアルタイム処理において、少なくとも、タスクが実行中、待機中、中断中のいずれであることを示す状態フラグ、タスクに実行要求があるか否かを示すタスクステータス、各タスクの実行を再開する時の優先順位を示す再開優先順位データ、各タスクを起動する際のレジスタ値、アドレス値等を示す再開データからなるタスクデータ及びタスクの実行順序が予め登録された実行順序テーブルを備え、実行順序テーブルに登録されているタスクの実行順序に従い実行するタスクを決定する際に、順番の回ってきたタスクの前記タスクステータスを検査し、実行可能であれば該タスクの実行を行うことを特徴とするタスク切り換え方法。

【請求項 2】 タスクの実行要求時に、該タスクの実行が可能であれば、該タスクと同時に実行を行えないタスクのタスクステータスを実行要求無しにする等他のタスクデータを変更することにより、タスク間の制御を行えるようにしたことを特徴とする請求項 1 記載のタスク切り換え方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、工作機械等に組み込んで機械を制御するプログラムを細かいタスクに分割し、分割した複数のタスクを時分割方法により順次実行していく場合のタスク切り換え方法に関するものである。

【0002】

【従来の技術】 従来の時分割によるタスクの切り換え方法について一例を挙げて説明する。図 5 に示すタスクデータ 35 は、タスクの情報を示すデータであり、あるタスクの実行要求が発生すると、タスクデータ 35 を記憶する領域を確保し、該タスクの実行終了時に記憶領域を解放する。タスクデータ 35 は、タスクの実行中、中断中、待機中を示す状態フラグ 31、タスクの実行優先順位を示す優先順位データ 32、タスクを起動する際のレジスタ値、アドレス値等を示す再開データ 33、タスクの次に実行要求が発生したタスクのタスクデータの先頭アドレスを示す次タスクデータ先頭アドレス 34 等から構成されている。

【0003】 図 6 は前記タスクデータ 35 をコントロールするための起動待ちタスクコントロールブロック 38 とタスクデータ 35 a～35 d の関係を示した図である。起動待ちタスクコントロールブロック 38 は、各タスクデータ 35 を管理するためのデータバッファであり、タスクの実行要求により発生したタスクデータ 35 の中で一番早く発生したタスクデータ 35 の先頭アドレスを記憶する先頭タスクデータアドレス 36、タスクの実行要求により発生したタスクデータ 35 の中で一番遅く発生したタスクデータ 35 の先頭アドレスを記憶する最終タスクデータアドレス 37 から構成されている。

【0004】 あるタスクの実行が要求されると、該タスクのタスクデータ 35 を記憶する領域を確保し、該タスクのタスクデータ 35 を初期化する。そして確保したタスクデータ 35 の先頭アドレスを最終タスクデータアドレス 37 に記憶するとともに該タスクの前に実行要求の発生したタスクのタスクデータ 35 の次タスクデータ先頭アドレスにも記憶する。これにより、実行要求の発生したタスクのタスクデータ 35 は繋がられ、実行要求が発生しているタスクのタスクデータ 35 を管理することが可能となる。

【0005】 図 6 の場合は、各タスクの実行要求により、タスクデータ 35 a～35 d が発生し、タスクデータ 35 a～35 d のうち、一番早く発生したタスクデータは 35 a であり、タスクデータ 35 a の先頭アドレスが先頭タスクデータアドレス 36 に記憶されている。同様に、一番遅く発生したタスクデータは 35 d であり、タスクデータ 35 d の先頭アドレスが最終タスクデータアドレス 37 に記憶されている。タスクデータ 35 a の次に発生したタスクデータは 35 b であり、タスクデータ 35 a の次タスクデータ先頭アドレス 34 a にタスクデータ 35 b の先頭アドレスが記憶されている。次タスクデータ先頭アドレス 34 b には、タスクデータ 35 c の先頭アドレスが記憶されている。

【0006】 図 6 の状態において、あるタスクの実行要求が発生すると、起動待ちタスクコントロールブロック 38 及び各タスクデータ 35 a～35 e の関係は図 7 のようになる。図 7 において、35 e は新たなタスクの実行要求によって発生したタスクデータである。図 6 の状態において、新たにタスクの実行要求が発生すると、そのタスクデータ 35 e を記憶する領域を確保し、タスクデータ 35 e を初期化する。そして最終タスクデータアドレス 37 にタスクデータ 35 e の先頭アドレスを記憶する。さらに、該タスクの前に実行要求の発生したタスクのタスクデータ 35 d の次タスクデータ先頭アドレス 34 d にタスクデータ 35 e の先頭アドレスを記憶する。

【0007】 従来の時分割によるリアルタイム処理では、上記した各データを用いて、次のようにタスクの切り換えが行われる。

【0008】 図 4 は実行するタスクの切り換え時の処理手順の一例を示したフローチャートであり、タイマ割り込み等によるタスクの切り換え処理では、タスク中断処理 27、起動タスク決定処理 29、タスク起動処理 30 が順番に行われる。また、タスクの実行終了によるタスク切り換え処理では、タスク実行終了処理 28、起動タスク決定処理 29、タスク起動処理 30 が順番に行われることを示している。

【0009】 仮に、図 7 の状態においてタスクデータ 35 a のタスク実行中に、タイマ割り込み等により、タスクの切り換え要求が発生すると、最初にタスク中断処理

27が行われる。タスク中断処理27では、切り換った時点のレジスタ値、アドレス値等を再開データ33aに退避し、状態フラグ31aを中断中にセットし、起動タスク決定処理29を行う。起動タスク決定処理29では、各タスクデータ35の優先順位データ32a~32eを比較し、最も優先順位の高いタスクを次に起動するタスクに決定し、タスク起動処理30を行う。

【0010】仮に、起動タスク決定処理29の結果、タスクデータ35bの優先順位が最も高かった場合、タスク起動処理30では、決定したタスクの再開データ33b等を復帰し、さらに、状態フラグ31bを実行中に変更して、決定したタスクの起動を行う。

【0011】また、タスクの実行終了により、タスク切り換え要求が発生すると、最初にタスク実行終了処理28が行われる。タスク実行終了処理28では、実行が終了したタスクのタスクデータ35の記憶領域を解放し、それに伴うタスクデータ35の変更、起動待ちタスクコントロールブロック38の変更等を行う。仮に、図7の状態において、タスクデータ35bの実行終了により、タスク切り換え要求が発生すると、タスク実行終了処理28では、タスクデータ35bの記憶領域を解放し、次タスクデータ先頭アドレス34aにタスクデータ35cの先頭アドレスを記憶する(図8)。そして上記と同様に起動タスク決定処理29、タスク起動処理30を順番に行う。

【0012】

【発明が解決しようとする課題】上記したように従来のリアルタイム処理では、起動待ちタスクコントロールブロック38の制御、タスクデータ35を記憶する領域の確保及び解放、タスクの実行要求時の処理、再開データ33の退避及び復帰、起動タスク決定処理29は実行するタスクを切り換える際に必ず行われる。

【0013】しかし、上記処理に要する時間は、システム全体から見た場合、無駄な時間である。また、上記したタスク切り換え方法では、優先順位が高く処理の長いタスクが実行されると、優先順位が同等レベル以下のタスクが長時間実行されない恐れがある。

【0014】本発明の目的は、上記した従来技術の欠点をなくし、実行要求タスクの登録、決定に要する時間を短縮するとともにタスクの制御を容易に行えるようにすることである。

【0015】

【課題を解決するための手段】工作機械等に組み込んで機械を制御するプログラムは、パソコン等を用いてアプリケーションプログラムを起動するプログラムと違って、システム作成時に予め実行するタスクは決まっている。また、システム作成時に予め実行するタスクは決まっているため、タスクの基本的な実行順序も予め決定することができる。従って、全タスクのタスクデータを予め設けておくことにより、起動待ちタスクコントロール

ブロックの制御、タスクデータ記憶領域の確保及び解放の処理を省くことができる。さらに、タスクの基本的な実行順序を示す実行順序テーブルの内容を予め設定しておき、実行順序テーブルに従ってタスクの実行順序を決定することにより、タスク切り換え時に毎回必ず行っていた優先順位の比較、起動タスク決定処理に要する時間を削減することが可能になるとともに特定タスクの長時間連続処理をなくすることが可能になる。

【0016】タスクの実行要求の有無を示すタスクステータスを各タスクに設け、タスクステータスを実行要求有りにすることにより、タスクの実行要求が発生したことにより、実行順序テーブルに従って順番が回ってきたときに、タスクステータスを検査し、タスクの起動を行うか否かを決定することによりタスクの実行要求に要する時間も減少させることができる。

【0017】また、各タスク間の排他、2つ以上のタスクを同時に実行する等タスク間の関係もシステム作成時には決定している。従って、各タスク間の関係を示す関係マトリックステーブルを設け、タスクの実行要求時に関係マトリックステーブルに基づいて、各タスクのタスクステータスを変更することにより、タスク間の制御も容易になる。

【0018】

【作用】この方法では、タスクの実行要求、実行するタスクの決定に要する時間を短縮することができるとともに優先順位が同等レベル以下のタスクの実行が可能になる。

【0019】

【実施例】以下実施例図面を参照して本発明タスク切り換え方法を用いたリアルタイム処理を説明する。図1は本発明タスク切り換え方法の処理手順の一実施例を示すフローチャートである。図2及び図9、図10は本発明タスク切り換え方法で使用するタスクデータ21及び実行順序テーブル25、関係マトリックステーブル26を示す。

【0019】図9の実行順序テーブル25には、カウンタ値1, 2, 3...Cnに対応して実行するタスク名を予め登録している。タスクカウンタには、カウンタ値1~Cnのいずれかを記憶しており、タスクカウンタに記憶されているカウンタ値に対応したタスクが次に実行するタスクの最優先候補となる。

【0020】図2はタスクデータ21であり、予め各タスクに必ず1個設けられている。タスクデータ21はタスクの情報を示すデータであり、タスクの実行中、中断中、待機中を示す状態フラグ17、タスクの実行を再開するときの優先順位を示す再開優先順位データ18、タスクを起動する際のレジスタ値、アドレス値等を示す再開データ19及びタスクの実行要求の有無を示すタスクステータス20から構成されている。

【0021】図10の関係マトリックステーブル26に

は、実行要求を発生するタスクに対応して、制御を行うタスクの処理がタスクごとに予め登録されている。例えばタスク1の実行を要求する場合には、図10の関係マトリックステーブル26に従ってタスク2～タスクnのタスクデータにS12～S1nの処理を行う。該処理とは、相手タスクのタスクステータス20、再開データ19等を変更することを指し、これらのデータを変更することにより、相手タスクの実行要求を行ったり、実行要求を取り消したりする。

【0022】最初にタスクの実行要求を発行する方法について述べる。図3はタスクの実行要求処理を示すフローチャートである。あるタスクの実行を要求する場合、タスクのタスクステータス20を実行要求有りにセットし、図10に示す関係マトリックステーブル26に基づいて、他のタスクのタスクデータ21を変更する。

【0023】次に、タスクの切り換え方法について図1のフローチャートを参照して述べる。タイマ等によりタスクの切り換え要求が発生すると、タスク中断処理1、起動タスク決定処理3、タスク起動処理4の順に各処理を行いタスクの切り換えを行う。また、タスクの実行終了により、タスクの切り換え要求が発生すると、タスク実行終了処理2、起動タスク決定処理3、タスク起動処理4の順に各処理を行いタスクの切り換えを行う。

【0024】タイマ割り込み等により実行タスクの切り換え要求が発生すると、タスク中断処理1を行う。タスク中断処理1では、実行していたタスクの状態フラグ17を中断中にセットし、再開データ19を待避する。これにより、再びこのタスクを実行するときに待避中の再開データ19を復帰させることにより、該タスクを中断した状態から引き続いて実行させることができる。

【0025】タスク中断処理1終了後、起動タスク決定処理3を行う。起動タスク決定処理3ではタスクカウンタを更新し、実行順序テーブル25に示すカウンタ値に対応したタスクを選択する。そして選択したタスクのタスクステータス20が実行要求有りであれば、該タスクを次に実行するタスクに決定する。該タスクが実行要求無しであれば、各タスクの状態フラグ17をチェックして実行中断中のタスクを検索する。実行中断中のタスクがあれば、その再開優先順位データ18を検査し、一番優先順位の高いタスクを次に実行するタスクに決定する。中断中のタスクが無かった場合、予め決めておいた補助タスクを次に実行するタスクに決定する。なお、補助タスクとは実行要求を発生しているタスクが無い場合に実行するタスクを指す。

【0026】起動タスク決定処理3終了後、タスク起動処理4を行う。タスク起動処理4では、状態フラグ17を実行中に変更し、再開データ19を復帰させ決定したタスクの実行を行う。タスクの実行終了により実行タス

クの切り換え要求が発生すると、タスク実行終了処理2を行う。タスク実行終了2処理では、実行していたタスクのタスクデータ21を初期化する。タスク実行終了処理2終了後、起動タスク決定処理3を行う。起動タスク決定処理3では、各タスクの状態フラグ17をチェックして、実行中断中のタスクを検索する。そして、実行中断中のタスクのうち、一番優先順位の高いタスクを次に実行するタスクに決定する。中断中のタスクが無かった場合、予め決めておいた補助タスクを次に実行するタスクに決定する。起動タスク決定処理3終了後、タスク起動処理4を行う。

【0027】

【発明の効果】本発明によれば、予め実行するタスクの順序を決めているため、実行要求タスクの登録及び実行するタスクの決定に要する時間を短縮することができる。とともにタスクの制御を容易に行うことができる。

【図面の簡単な説明】

【図1】本発明タスク切り換え方法の一実施例を示すフローチャート。

【図2】本発明で使用されるタスクデータの一例を示すデータ図。

【図3】本発明のタスク実行要求処理を示すフローチャート。

【図4】本発明タスク切り換え方法の処理手順の一例を示すフローチャート。

【図5】従来のタスク切り換え方法で使用されるタスクデータの一例を示すデータ図。

【図6】従来の起動待ちタスクコントロールブロックとタスクデータの関係を示すデータ図。

【図7】図6の状態から変化した状態を示すデータ図。

【図8】図7の状態から変化した状態を示すデータ図。

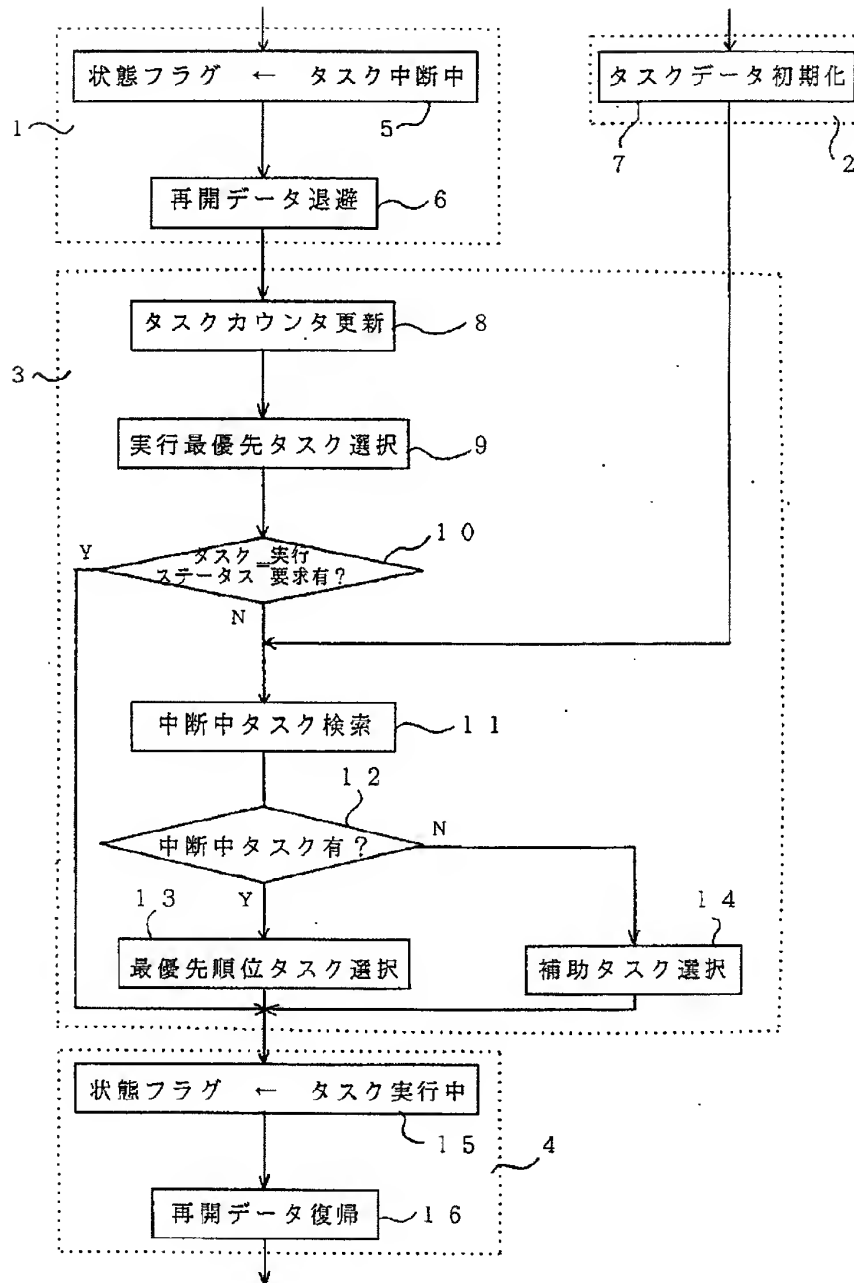
【図9】本発明タスク切り換え方法で使用される実行順序テーブルを示すテーブル。

【図10】本発明タスク切り換え方法で使用される関係マトリックステーブルを示すテーブル。

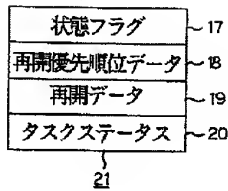
【符号の説明】

1, 27…タスク中断処理、2, 28…タスク実行終了処理、3, 29…起動タスク決定処理、4, 30…タスク起動処理、17, 31, 33a～33e…状態フラグ、18, 32, 33a～33e…再開優先順位データ、19, 33, 33a～33e…再開データ、20…タスクステータス、21, 35, 35a～35e…タスクデータ、25…実行順序テーブル、26…関係マトリックステーブル、34, 34a～34e…次タスクデータ先頭アドレス、36…先頭タスクデータアドレス、37…最終タスクデータアドレス、38…起動待ちタスクコントロールブロック。

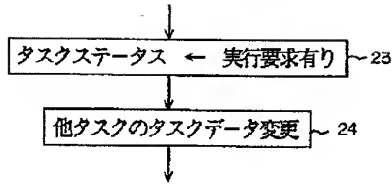
【図1】



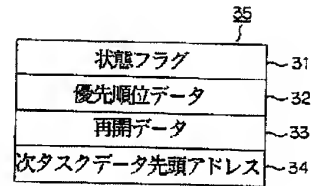
【図 2】



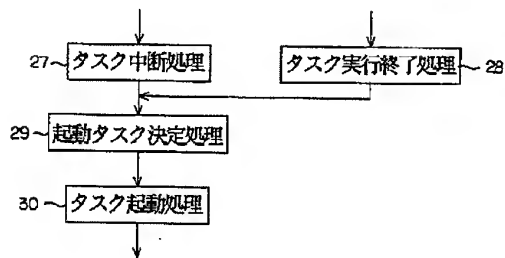
【図 3】



【図 5】



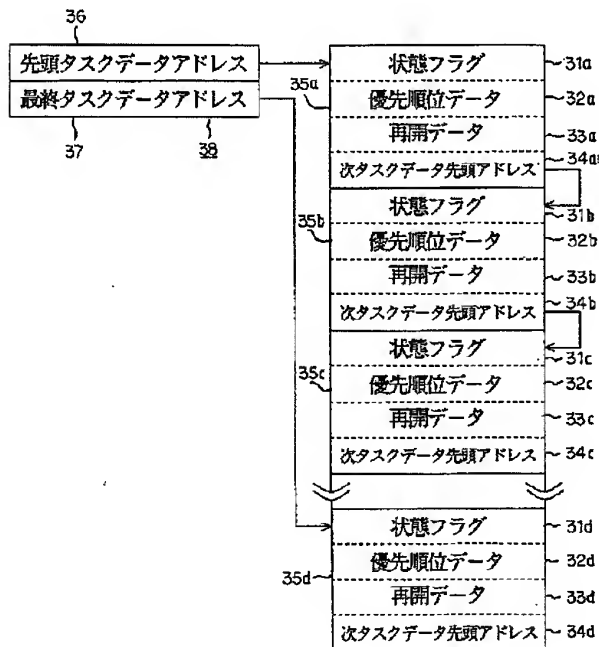
【図 4】



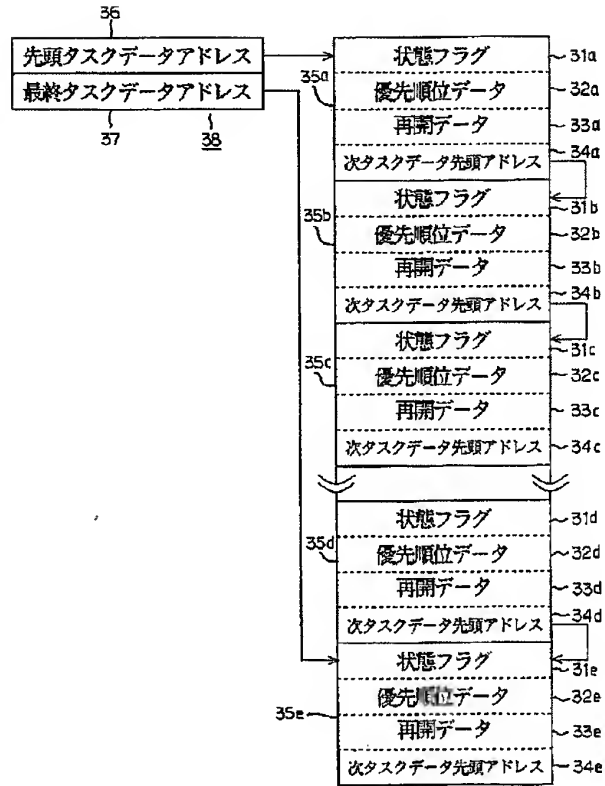
【図 9】

タスクカウンタ のカウンタ値	実行タスク名
1	タスク 1
2	タスク 2
3	タスク 3
⋮	⋮
Cn	タスク n

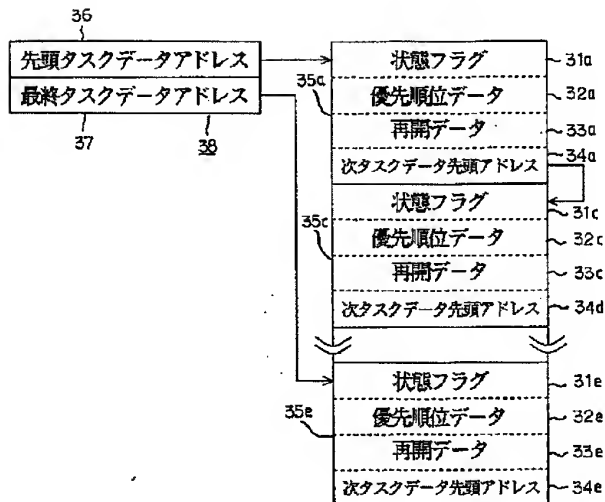
【図 6】



【図7】



【図8】



(8)

特開平8-190491

【図10】

26
}

タスク名 実行タスク名	タスク 1	タスク 2	タスク 3	...	タスク n
タスク 1		S 1 2	S 1 3	..	S 1 n
タスク 2	S 2 1		S 2 3	..	S 2 n
タスク 3	S 3 1	S 3 2		..	S 3 n
.
.
タスク n	S n 1	S n 2	S n 3	..	